



## SECS1024 - Laboratoire 10 - Android Diva

laboratoire noté 8 points - 10% de la note finale

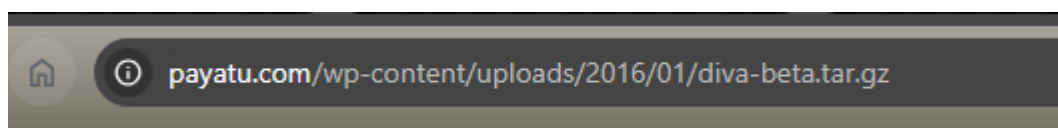
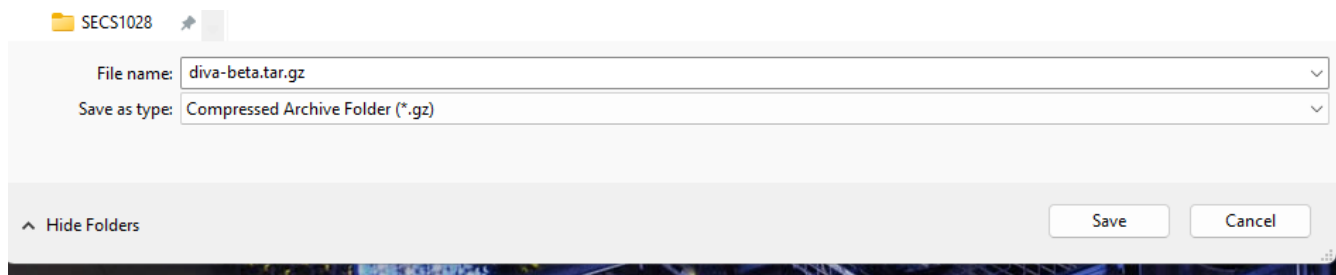
à rendre pour le 1er avril

Objectif : trouver des failles de sécurité sur une application Android. Utilisez pour cela une analyse statique (le code source de l'application est visible sur <https://github.com/payatu/divaandroid>) et une analyse dynamique sur Android-x86 (en réseau interne de VirtualBox).

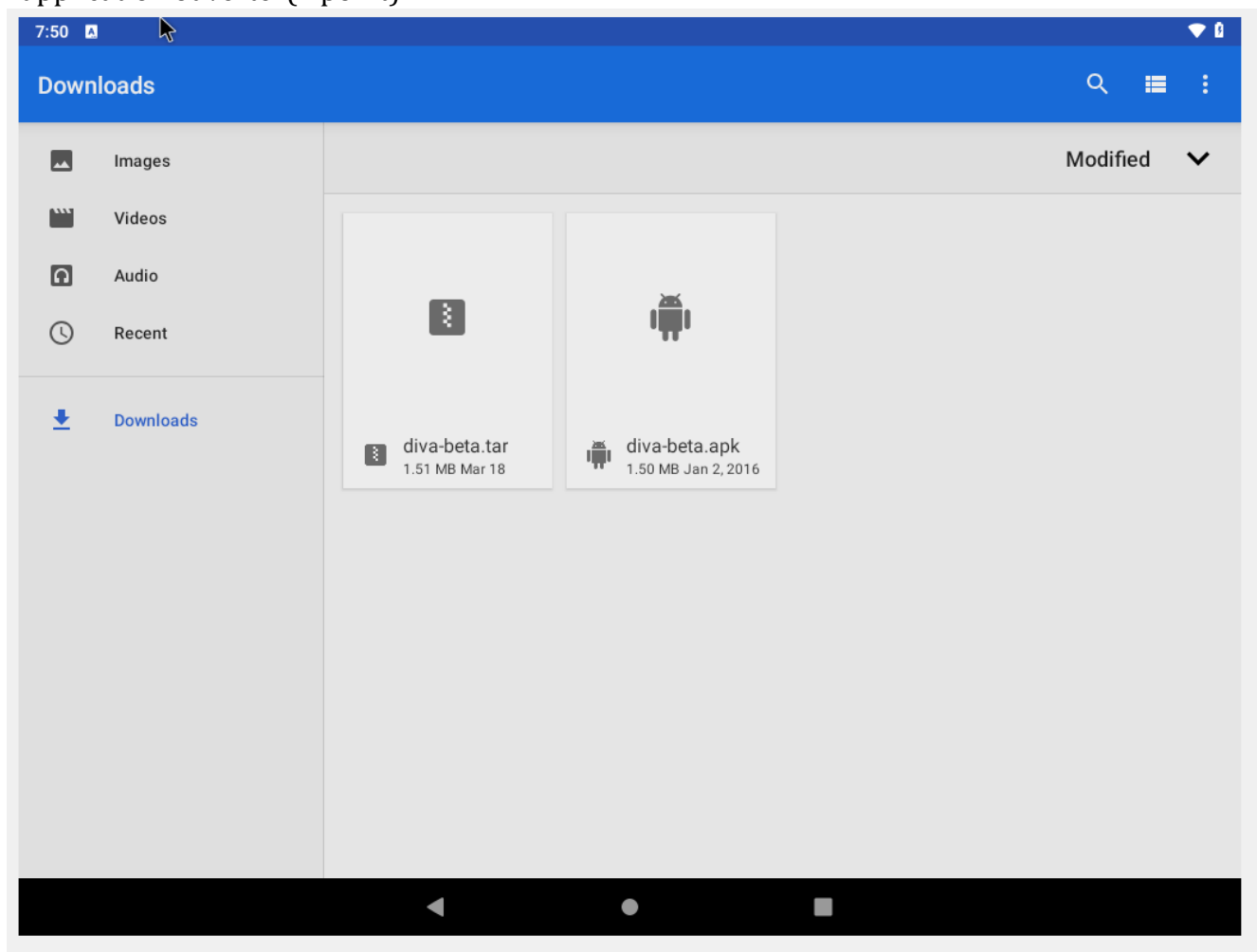
Mise en place du laboratoire : utilisez pour ce laboratoire une VM Android-x86 connectée réseau en interne sur VirtualBox.

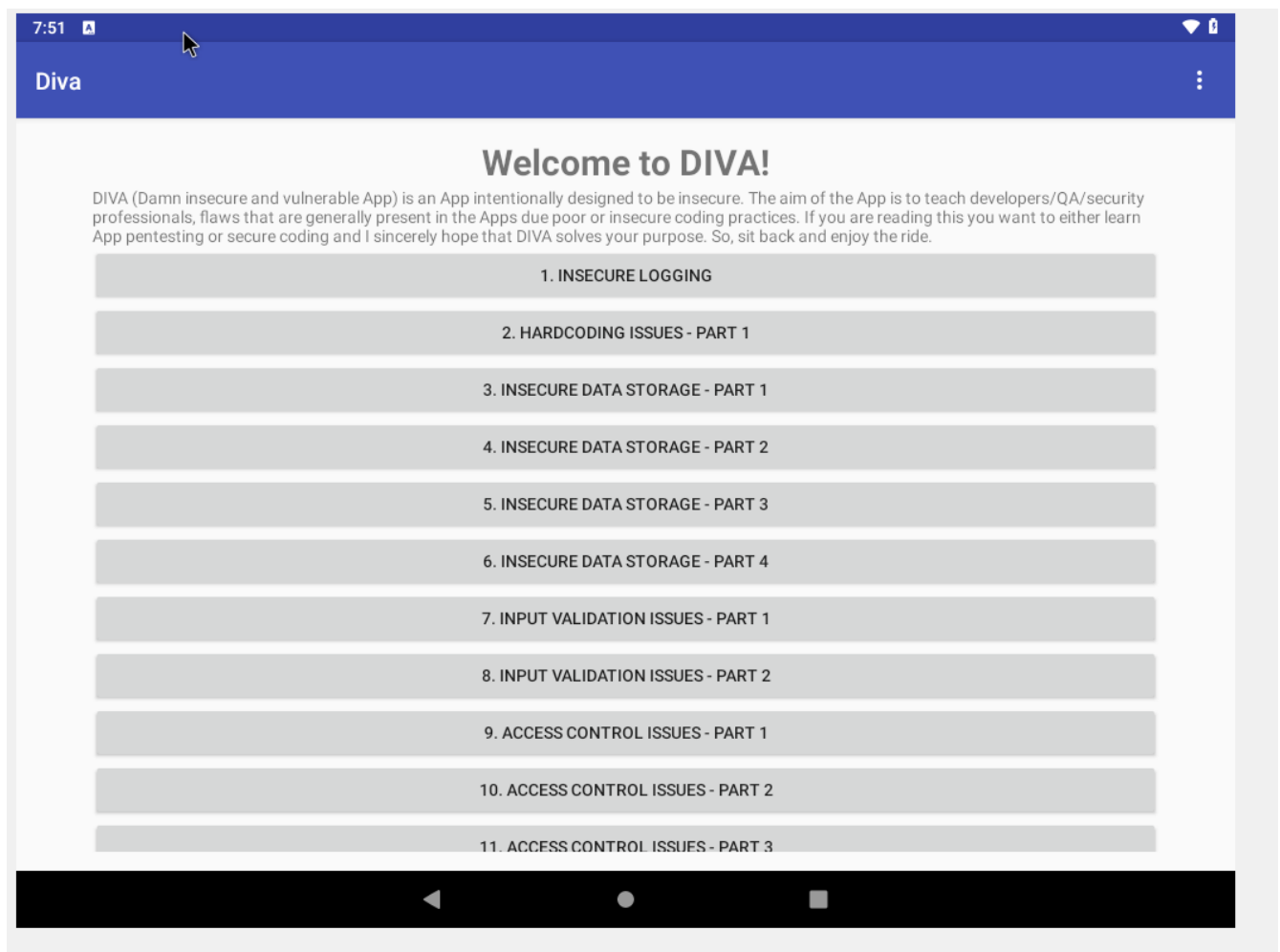
### 1 Installation de DIVA

(1 point) Diva est une application Android volontairement vulnérable. Elle est téléchargeable sur: <http://www.payatu.com/wp-content/uploads/2016/01/diva-beta.tar.gz>



- 1) installez cette application sur Android-x86 (en mode NAT) et faites une capture d'écran de l'application ouverte. (1 point)





## 2 Problèmes de sécurité (7 points)

Connectez l'application Diva sur le réseau interne VB (hors Internet).

Recherchez 7 failles de sécurité parmi les 13 problèmes/menus proposés.

Documentez vos réponses par des explications textuelles et des captures d'écran. (1 point par problème).

#1 insecure logging

8:05



## 1. Insecure Logging

**Objective:** Find out what is being logged where/how and the vulnerable code.

**Hint:** Insecure logging occurs when developers intentionally or unintentionally log sensitive information such as credentials, session IDs, financial details etc.

123456

CHECK OUT

```
Trusted GRUB now booting 'Android-x86 9.0-r2'
```

```
Progress: [██████████] Detecting Android-x86... found at /dev/sda1
```

```
console:/ # logcat | grep 123456
```

```
^C
```

```
130|console:/ # logcat | grep 123456
```

```
03-25 08:05:48.173 7321 7321 E diva-log: Error while processing transaction with credit card: 123456
```

```
03-25 08:05:49.122 7321 7321 E diva-log: Error while processing transaction with credit card: 123456
```

#2 hardcoding issues

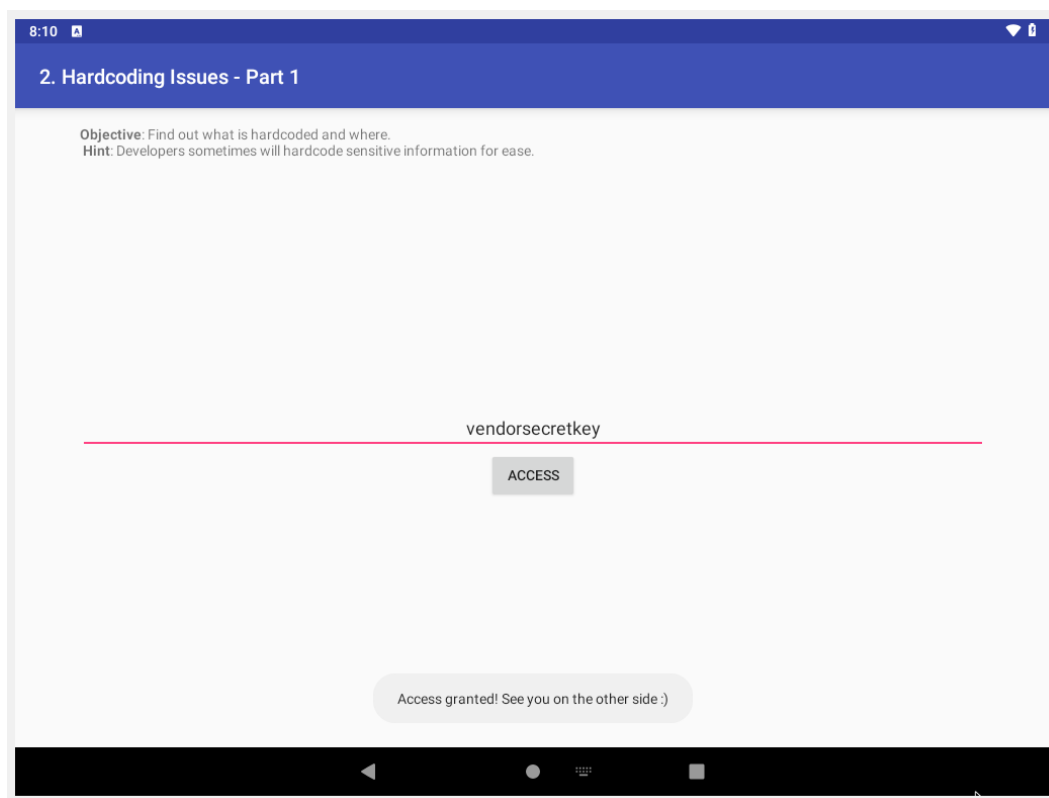
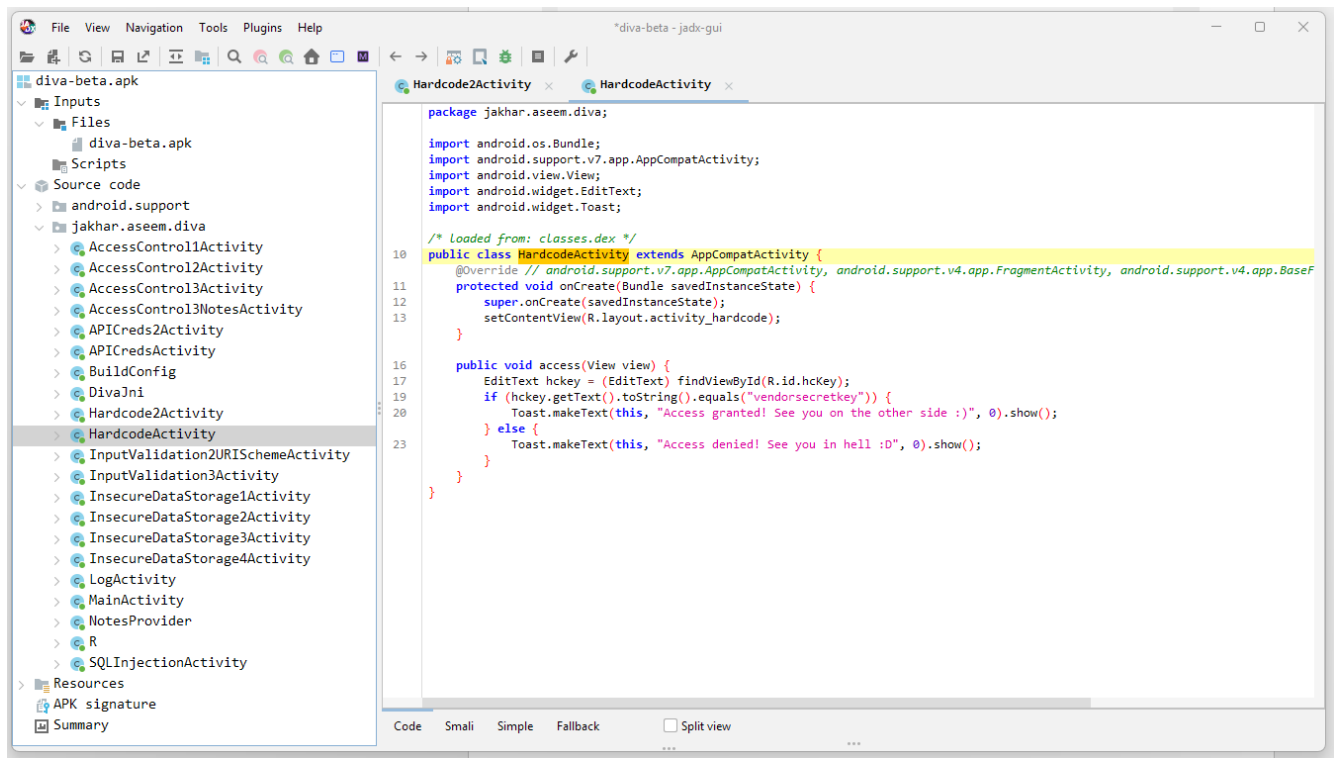
## 2. Hardcoding Issues - Part 1

**Objective:** Find out what is hardcoded and where.

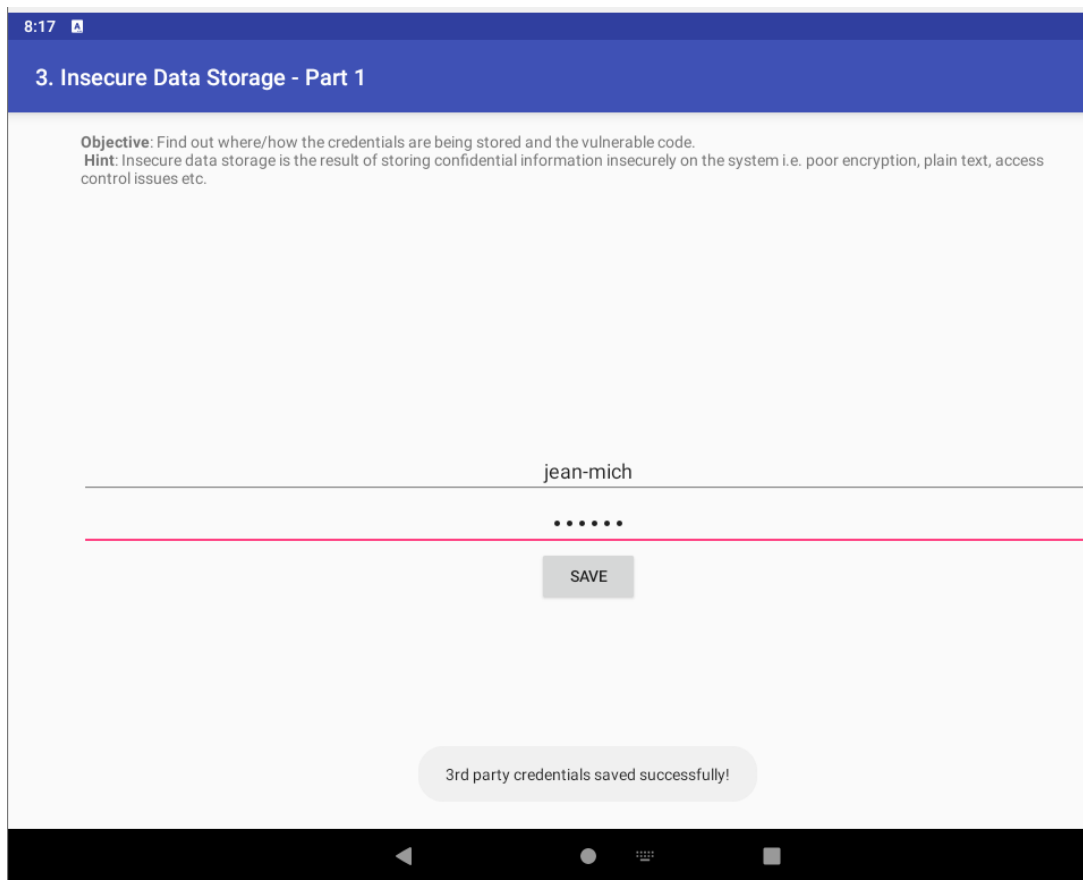
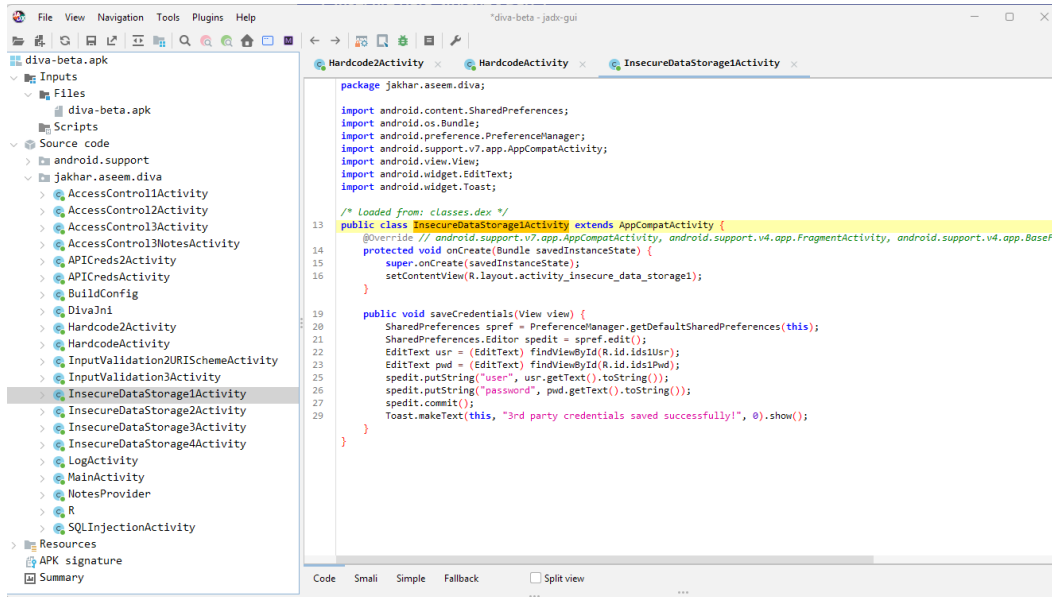
**Hint:** Developers sometimes will hardcode sensitive information for ease.

Enter the Vendor key

ACCESS



### #3 insecure data storage part 1



```
console:/data/data # cd /
console:/ # cd data
console:/data # cd data
console:/data/data # cd jakhar.aseem.diva/
console:/data/data/jakhar.aseem.diva # ls
cache code_cache databases
console:/data/data/jakhar.aseem.diva # ls
cache code_cache databases shared_prefs
console:/data/data/jakhar.aseem.diva # cd shared_prefs/
console:/data/data/jakhar.aseem.diva/shared_prefs # ls
jakhar.aseem.diva_preferences.xml
console:/data/data/jakhar.aseem.diva/shared_prefs # cat jakhar.aseem.diva_preferences.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
  <string name="password">pierre</string>
  <string name="user">jean-mich</string>
</map>
console:/data/data/jakhar.aseem.diva/shared_prefs # _
```

#### #4 insecure data storage part 2

8:22

4. Insecure Data Storage - Part 2

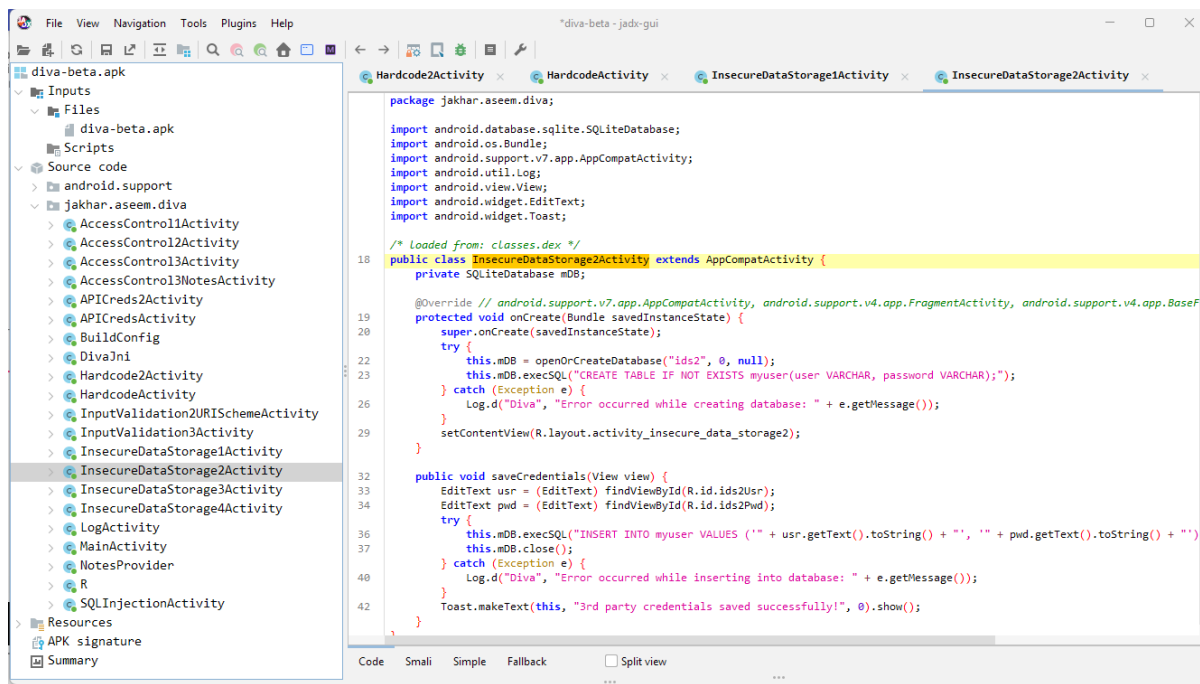
**Objective:** Find out where/how the credentials are being stored and the vulnerable code.  
**Hint:** Insecure data storage is the result of storing confidential information insecurely on the system i.e. poor encryption, plain text, access control issues etc.

jean-mich

.....

SAVE





```
console:/data/data/jakhar.aseem.diva/databases # cd ../
console:/data/data/jakhar.aseem.diva # cd databases/
console:/data/data/jakhar.aseem.diva/databases # ls
divanotes.db divanotes.db-shm divanotes.db-wal ids2
console:/data/data/jakhar.aseem.diva/databases # vi ids2
```



#5 insecure data storage part 3

## 5. Insecure Data Storage - Part 3

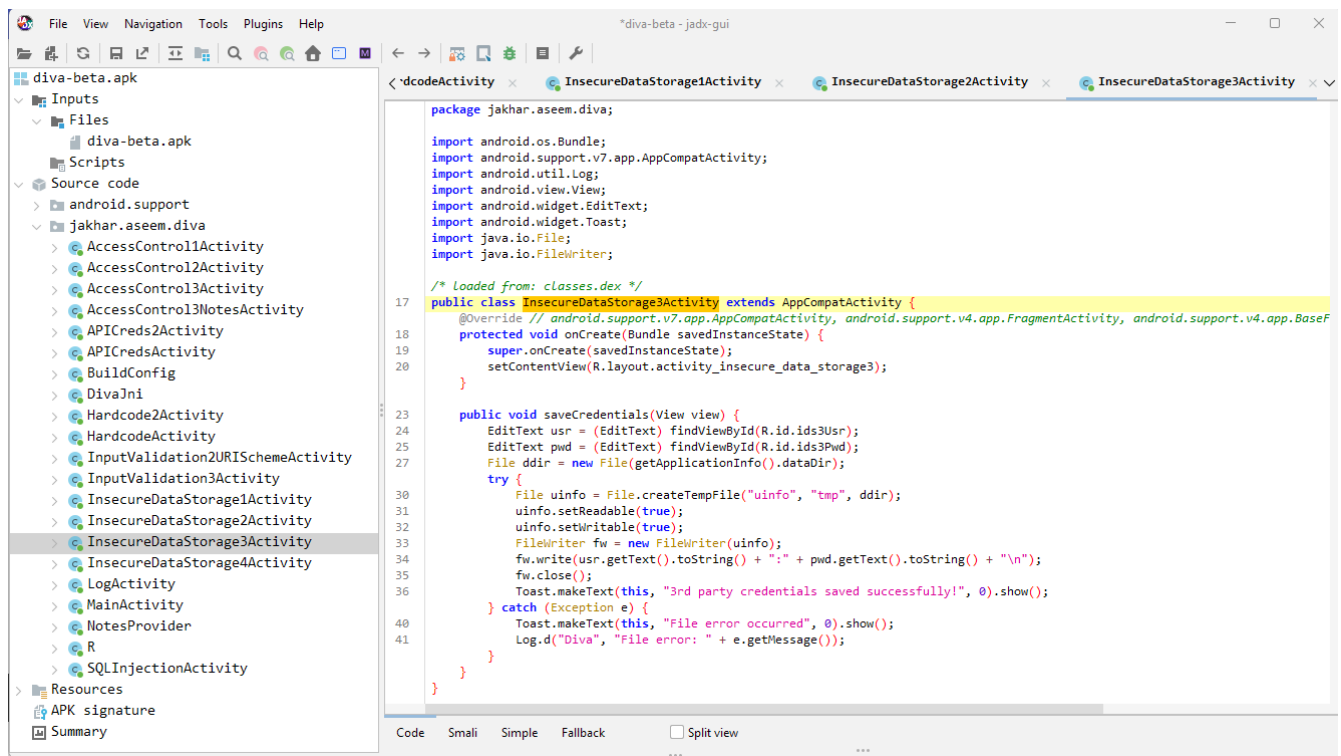
**Objective:** Find out where/how the credentials are being stored and the vulnerable code.

**Hint:** Insecure data storage is the result of storing confidential information insecurely on the system i.e. poor encryption, plain text, access control issues etc.

pierrepaul

.....

SAVE



```

[3] + Stopped          \vi ids2
console:/data/data/jakhar.aseem.diva/databases # cd ../
console:/data/data/jakhar.aseem.diva # cd databases/
console:/data/data/jakhar.aseem.diva/databases # ls
divanotes.db divanotes.db-shm divanotes.db-wal ids2
console:/data/data/jakhar.aseem.diva/databases # cd ../
console:/data/data/jakhar.aseem.diva # ls
cache code_cache databases shared_prefs uinfo116345773485917010tmp
console:/data/data/jakhar.aseem.diva # cat
cache/                  databases/                  uinfo116345773485917010tmp
code_cache/            shared_prefs/
console:/data/data/jakhar.aseem.diva # cat
cache/                  databases/                  uinfo116345773485917010tmp
code_cache/            shared_prefs/
console:/data/data/jakhar.aseem.diva # cat uinfo116345773485917010tmp
pierrepaullamirande
console:/data/data/jakhar.aseem.diva #

```

## #6 insecure data storage

## 6. Insecure Data Storage - Part 4

**Objective:** Find out where/how the credentials are being stored and the vulnerable code.

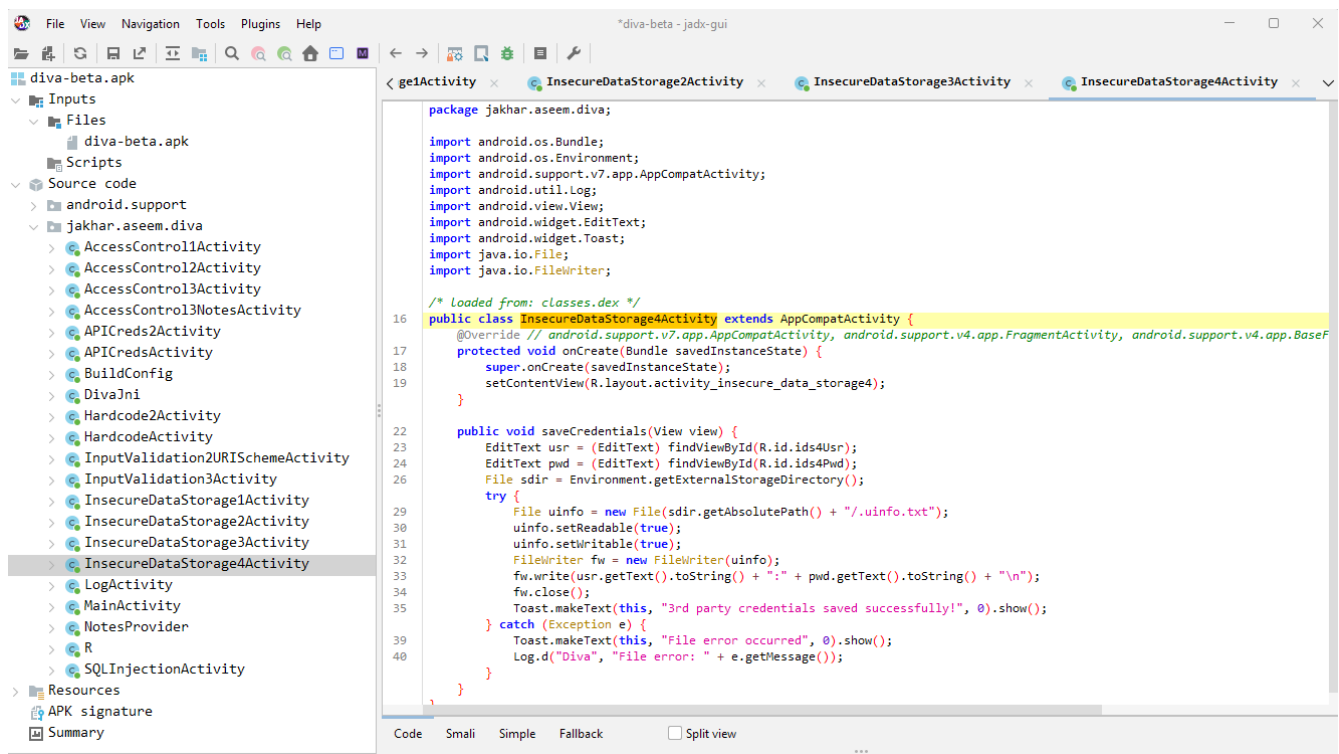
**Hint:** Insecure data storage is the result of storing confidential information insecurely on the system i.e. poor encryption, plain text, access control issues etc.

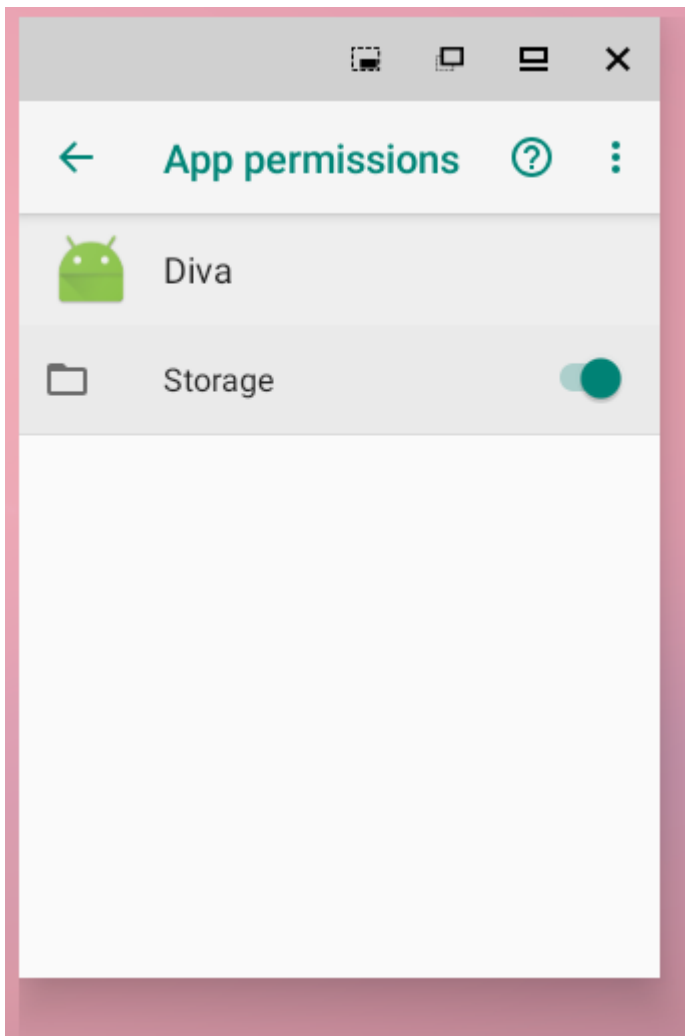
ginette

.....

SAVE

File error occurred







## 6. Insecure Data Storage - Part 4

**Objective:** Find out where/how the credentials are being stored and the vulnerable code.

**Hint:** Insecure data storage is the result of storing confidential information insecurely on the system i.e. poor encryption, plain text, access control issues etc.

ginette

.....

SAVE

3rd party credentials saved successfully!



9:  
3/



```
Android [Running] - Oracle VirtualBox
File Machine View Input Devices Help
console:/data/data/jakhar.aseem.diva # cd /
console:/ # cd sdcard
console:/sdcard # ls
Alarms Android DCIM Download Movies Music Notifications Pictures Podcasts Ringtones
console:/sdcard # ls -la
total 52
drwxr-x--x 12 root sdcard_rw 4096 2025-03-25 08:50 .
drwx--x--x 4 root sdcard_rw 4096 2025-03-11 09:40 ..
-rw-rw---- 1 root sdcard_rw 15 2025-03-25 08:50 .uinfo.txt
drwxr-x--x 2 root sdcard_rw 4096 2025-03-11 09:40 Alarms
drwxr-x--x 3 root sdcard_rw 4096 2025-03-11 09:40 Android
drwxr-x--x 2 root sdcard_rw 4096 2025-03-11 09:40 DCIM
drwxr-x--x 2 root sdcard_rw 4096 2025-03-18 09:56 Download
drwxr-x--x 2 root sdcard_rw 4096 2025-03-11 09:40 Movies
drwxr-x--x 2 root sdcard_rw 4096 2025-03-11 09:40 Music
drwxr-x--x 2 root sdcard_rw 4096 2025-03-11 09:40 Notifications
drwxr-x--x 2 root sdcard_rw 4096 2025-03-11 09:40 Pictures
drwxr-x--x 2 root sdcard_rw 4096 2025-03-11 09:40 Podcasts
drwxr-x--x 2 root sdcard_rw 4096 2025-03-11 09:40 Ringtones
console:/sdcard # cat .uinfo.txt
ginette:michel
console:/sdcard # _
```

## #7 input validation issues part 1

```
diva-beta - jadt-gui
File View Navigation Tools Plugins Help
diva-beta.apk
  Inputs
    Files
      diva-beta.apk
    Scripts
  Source code
    android.support
    jakhar.aseem.diva
      AccessControl1Activity
      AccessControl2Activity
      AccessControl3Activity
      AccessControl3NotesActivity
      APICreds2Activity
      APICredsActivity
      BuildConfig
      DivaJni
      HardCode2Activity
      HardCodeActivity
      InputValidation2URISchemeActivity
      InputValidation3Activity
      InsecureDataStorage1Activity
      InsecureDataStorage2Activity
      InsecureDataStorage3Activity
      InsecureDataStorage4Activity
      LoginActivity
      MainActivity
      NotesProvider
      R
      SQLInjectionActivity
  Resources
    APK signature
    Summary

<ataStorage2Activity x InsecureDataStorage3Activity x InsecureDataStorage4Activity x SQLInjectionActivity x
16 public class SQLInjectionActivity extends AppCompatActivity {
17     private SQLiteDatabase mDB;
18     @Override // android.support.v7.app.AppCompatActivity, android.support.v4.app.FragmentActivity, android.support.v4.app.BaseFra
19     protected void onCreate(Bundle savedInstanceState) {
20         super.onCreate(savedInstanceState);
21         try {
22             this.mDB = openOrCreateDatabase("sql", 0, null);
23             this.mDB.execSQL("DROP TABLE IF EXISTS sqluser;");
24             this.mDB.execSQL("CREATE TABLE IF NOT EXISTS sqluser(user VARCHAR, password VARCHAR, credit_card VARCHAR);");
25             this.mDB.execSQL("INSERT INTO sqluser VALUES ('admin', 'password123', '1234567812345678');");
26             this.mDB.execSQL("INSERT INTO sqluser VALUES ('diva', 'password', '11112223334444');");
27             this.mDB.execSQL("INSERT INTO sqluser VALUES ('john', 'password123', '5555666677778888');");
28         } catch (Exception e) {
29             Log.d("Divasql", "Error occurred while creating database for SQL: " + e.getMessage());
30         }
31         setContentView(R.layout.activity_sqlinjection);
32     }
33     public void search(View view) {
34         EditText srchtxt = (EditText) findViewById(R.id.ivlsearch);
35         try {
36             Cursor cr = this.mDB.rawQuery("SELECT * FROM sqluser WHERE user = '" + srchtxt.getText().toString() + "'", null);
37             StringBuilder strb = new StringBuilder("");
38             if (cr != null && cr.getCount() > 0) {
39                 cr.moveToFirst();
40                 do {
41                     strb.append("User: (" + cr.getString(0) + ") pass: (" + cr.getString(1) + ") Credit card: (" + cr.getString(2)
42                     ) while (cr.moveToNext());
43                 } else {
44                     strb.append("User: (" + srchtxt.getText().toString() + ") not found");
45                 }
46             }
47             Toast.makeText(this, strb.toString(), 0).show();
48         } catch (Exception e) {
49             Log.d("Divasql", "Error occurred while searching in database: " + e.getMessage());
50         }
51     }
52 }
```

## 7. Input Validation Issues - Part 1

**Objective:** Try to access all user data without knowing any user name. There are three users by default and your task is to output data of all the three users with a single malicious search.

**Hint:** Improper or no input validation issue arise when the input is not filtered or validated before using it. When developing components that take input from outside, always validate it. For ease of testing there are three users already present in the database, for example one of them is admin, you can try searching for admin to test the output.

pierre' or 1 != '1

SEARCH

User: (admin) pass: (passwd123) Credit card: (1234567812345678)  
User: (diva) pass: (p@ssword) Credit card: (1111222233334444)  
User: (john) pass: (password123) Credit card: (5555666677778888)

fin du laboratoire.